

WEB APPLICATION TESTING

http://www.tutorialspoint.com/software_testing_dictionary/web_application_testing.htm

Copyright © tutorialspoint.com

What is Web Application Testing?

Web application testing, a software testing technique exclusively adopted to test the applications that are hosted on web in which the application interfaces and other functionalities are tested.

Web Application Testing - Techniques:

1. **Functionality Testing** - The below are some of the checks that are performed but not limited to the below list:

- Verify there is no dead page or invalid redirects.
- First check all the validations on each field.
- Wrong inputs to perform negative testing.
- Verify the workflow of the system.
- Verify the data integrity.

2. **Usability testing** - To verify how the application is easy to use with.

- Test the navigation and controls.
- Content checking.
- Check for user intuition.

3. **Interface testing** - Performed to verify the interface and the dataflow from one system to other.

4. **Compatibility testing**- Compatibility testing is performed based on the context of the application.

- Browser compatibility
- Operating system compatibility
- Compatible to various devices like notebook, mobile, etc.

5. **Performance testing** - Performed to verify the server response time and throughput under various load conditions.

- **Load testing** - It is the simplest form of testing conducted to understand the behaviour of the system under a specific load. Load testing will result in measuring important business critical transactions and load on the database, application server, etc. are also monitored.
- **Stress testing** - It is performed to find the upper limit capacity of the system and also to determine how the system performs if the current load goes well above the expected maximum.
- **Soak testing** - Soak Testing also known as endurance testing, is performed to determine the system parameters under continuous expected load. During soak tests the parameters such as memory utilization is monitored to detect memory leaks or other performance issues. The main aim is to discover the system's performance under sustained use.
- **Spike testing** - Spike testing is performed by increasing the number of users suddenly by a very large amount and measuring the performance of the system. The main aim is to determine whether the system will be able to sustain the work load.

6. **Security testing** - Performed to verify if the application is secured on web as data theft and

unauthorized access are more common issues and below are some of the techniques to verify the security level of the system.

- Injection
- Broken Authentication and Session Management
- Cross-Site Scripting *XSS*
- Insecure Direct Object References
- Security Misconfiguration
- Sensitive Data Exposure
- Missing Function Level Access Control
- Cross-Site Request Forgery *CSRF*
- Using Components with Known Vulnerabilities
- Unvalidated Redirects and Forwards

Loading [MathJax]/jax/output/HTML-CSS/jax.js